LLaVA: Connects a Vision encoder and an LLM for general purpose visual and language understanding.

Instruction tuning:

Visual Instruction Data Generation:

1) Many multimodal datasets such as CC, LAION, COCO.. But, not necessarily an instruction-response pairs
2) Take image-caption pairs or image bounding box pairs, and feed the text to GPT-4. Then they ask the GPT-4 to generate a bunch of questions and answers from the captions or the bounding box.

   They ask GPT-4 to give conversations, detailed descriptions and complex reasoning task outcomes.
3) So, with the normal multimodal datasets like above and application of GPT on datasets like these, they have generated new datasets that help us to do instruction tuning.

Visual Instruction Tuning Model:

- Language Transformer (Vicuna)
- Vision Transformer (CLIP with ViT-L/14)
- Add a projection layer in between to put the image in the same space as the text token.

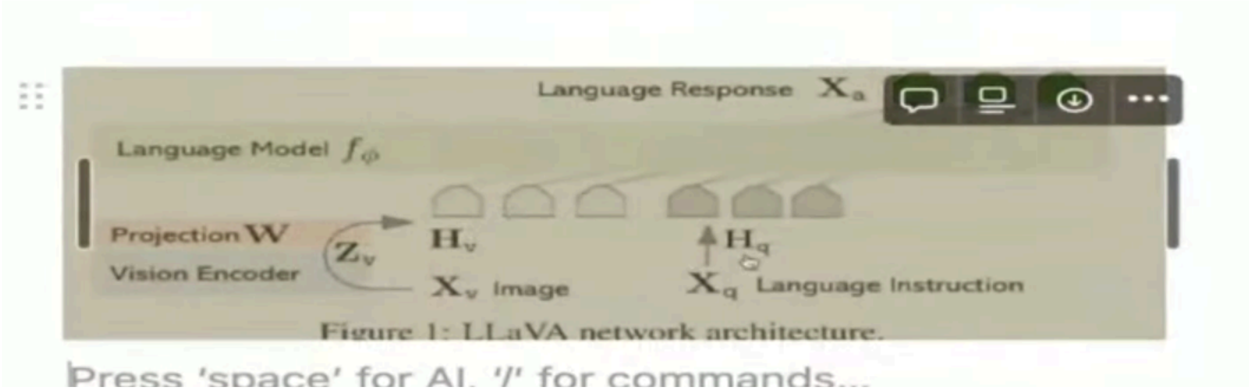
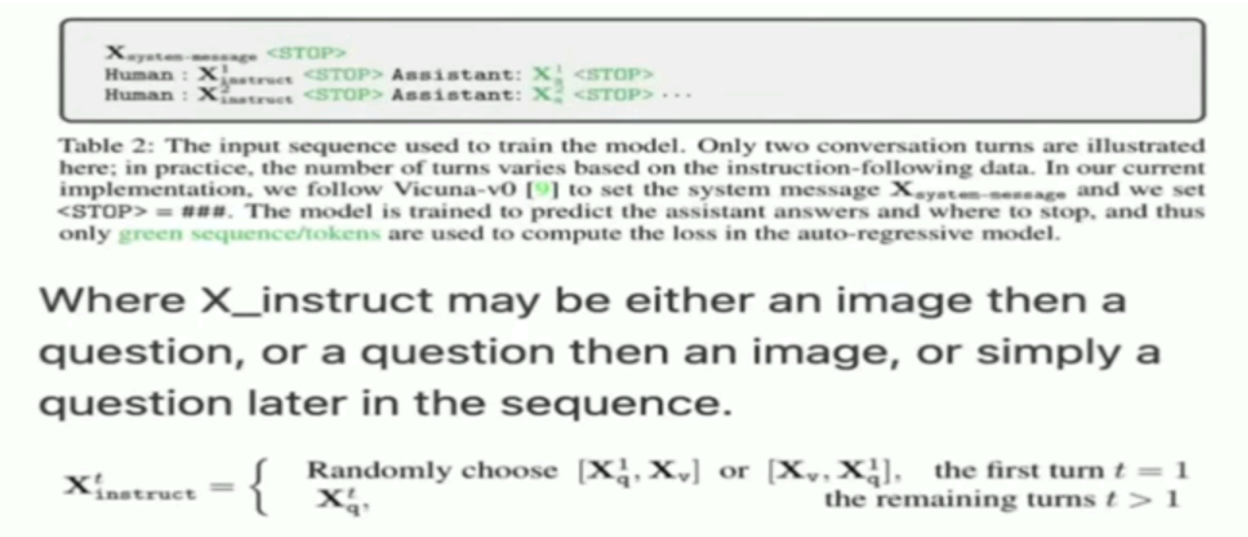


Figure 1: LLaVA network architecture.

- They are keeping the vision encoder and the language model frozen and are only training W.
- Then they organize their prompts into a simple conversation format



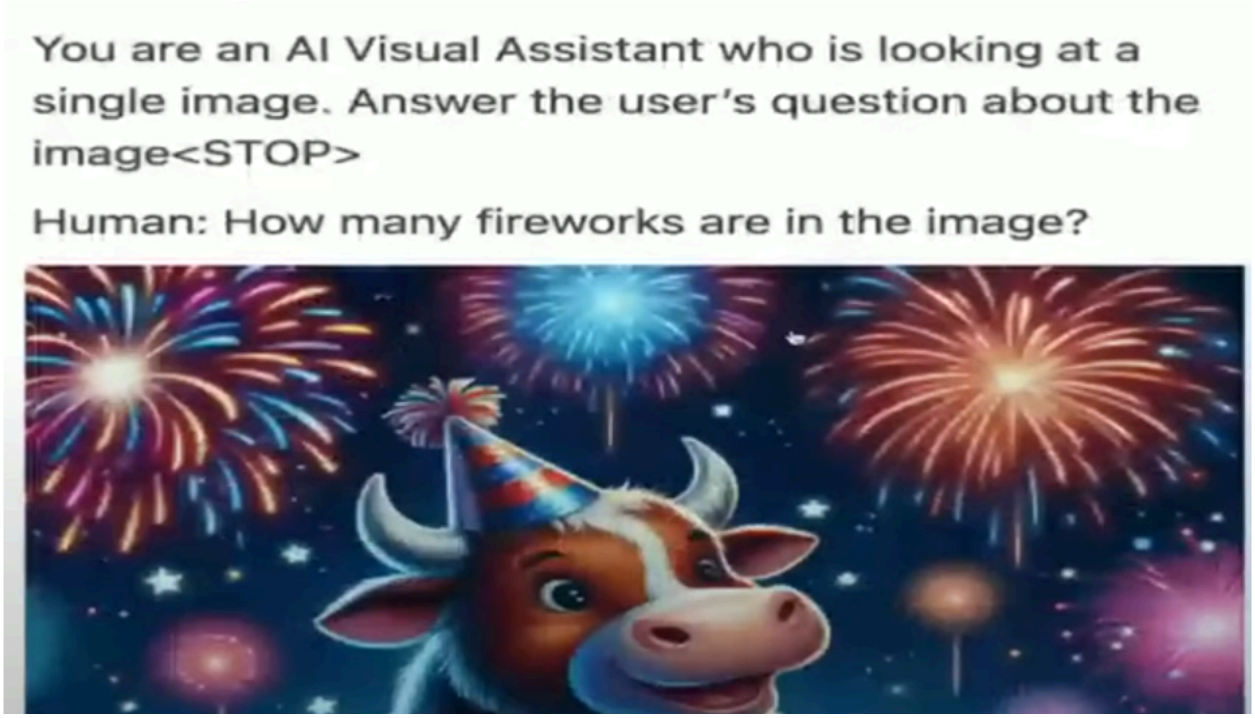
$\mathbf{X}_{\texttt{system-message}}$ <STOP>
Human : $\mathbf{X}^1_{\texttt{instruct}}$ <STOP> Assistant: $\mathbf{X}^1_{\texttt{a}}$ <STOP>
Human : $\mathbf{X}^2_{\texttt{instruct}}$ <STOP> Assistant: $\mathbf{X}^2_{\texttt{a}}$ <STOP> $\cdots$

Table 2: The input sequence used to train the model. Only two conversation turns are illustrated here; in practice, the number of turns varies based on the instruction-following data. In our current implementation, we follow Vicuna-v0 [9] to set the system message $\mathbf{X}_{\texttt{system-message}}$ and we set <STOP> = ###. The model is trained to predict the assistant answers and where to stop, and thus only green sequence/tokens are used to compute the loss in the auto-regressive model.

Where X_instruct may be either an image then a question, or a question then an image, or simply a question later in the sequence.

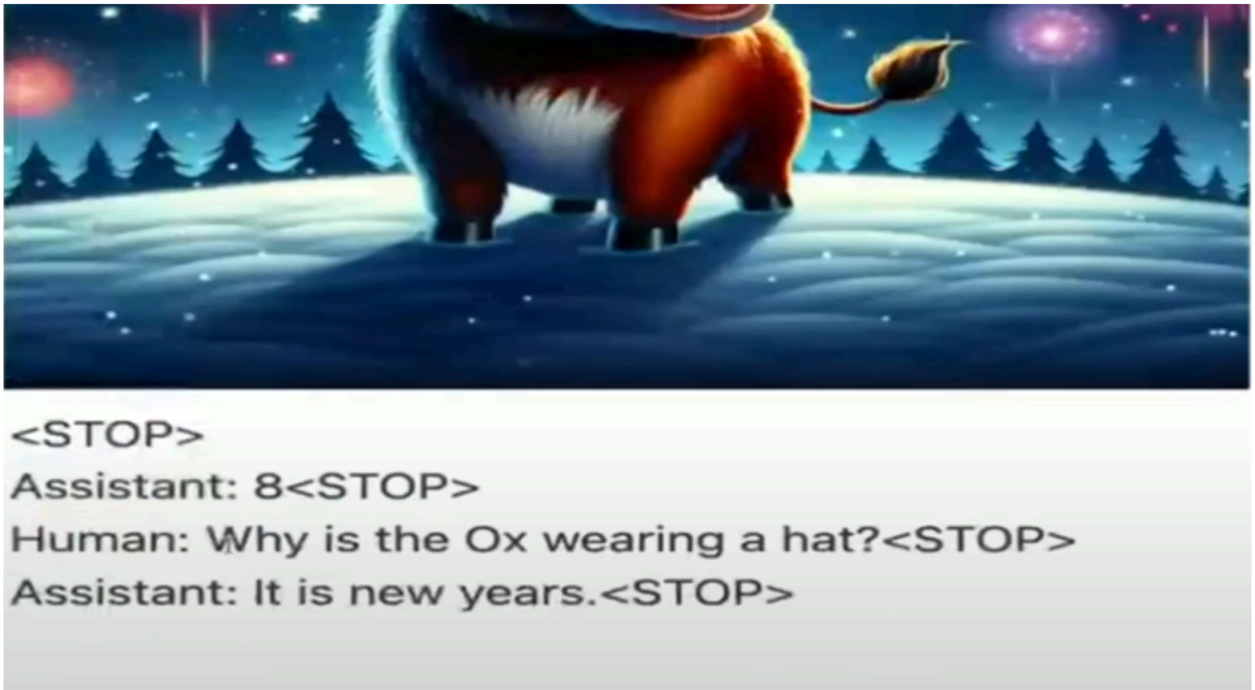
$$\mathbf{X}^t_{\texttt{instruct}} = \begin{cases} \text{Randomly choose } [\mathbf{X}^1_{\text{q}}, \mathbf{X}_{\text{v}}] \text{ or } [\mathbf{X}_{\text{v}}, \mathbf{X}^1_{\text{q}}], & \text{the first turn } t = 1 \\ \mathbf{X}^t_{\text{q}}, & \text{the remaining turns } t > 1 \end{cases}$$

- The vision transformer and language model are trained on different datasets and now with the projection layer between them.
- Example prompt -



You are an AI Visual Assistant who is looking at a single image. Answer the user's question about the image<STOP>

Human: How many fireworks are in the image?

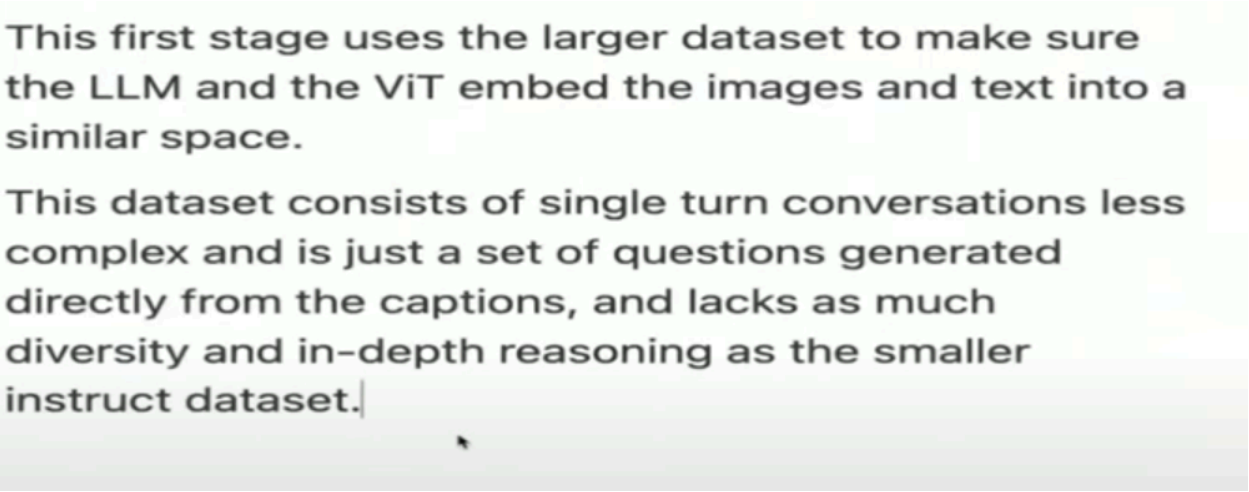


<STOP>
Assistant: 8<STOP>
Human: Why is the Ox wearing a hat?<STOP>
Assistant: It is new years.<STOP>

Process:

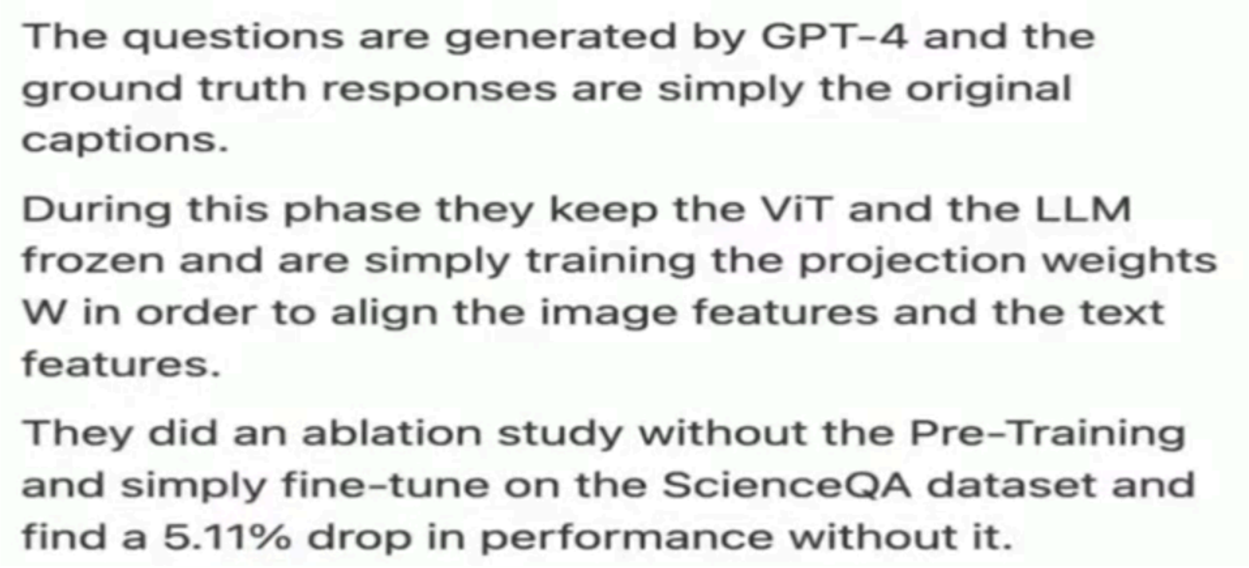
Stage 1: Pre-training for Feature Alignment.

They perform a larger pre-training on a subset of CC3M dataset which is filtered down to 595k image-text pairs.



This first stage uses the larger dataset to make sure the LLM and the ViT embed the images and text into a similar space.

This dataset consists of single turn conversations less complex and is just a set of questions generated directly from the captions, and lacks as much diversity and in-depth reasoning as the smaller instruct dataset.



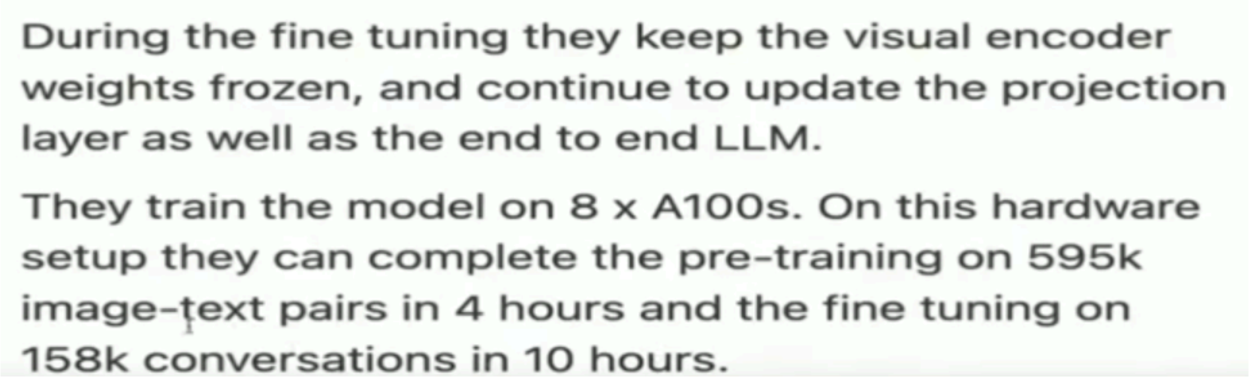
The questions are generated by GPT-4 and the ground truth responses are simply the original captions.

During this phase they keep the ViT and the LLM frozen and are simply training the projection weights W in order to align the image features and the text features.

They did an ablation study without the Pre-Training and simply fine-tune on the ScienceQA dataset and find a 5.11% drop in performance without it.

They say that if they have not pre-trained and directly fine tuned on the ScienceQA dataset as mentioned above, they've observed a 5% drop in accuracy.

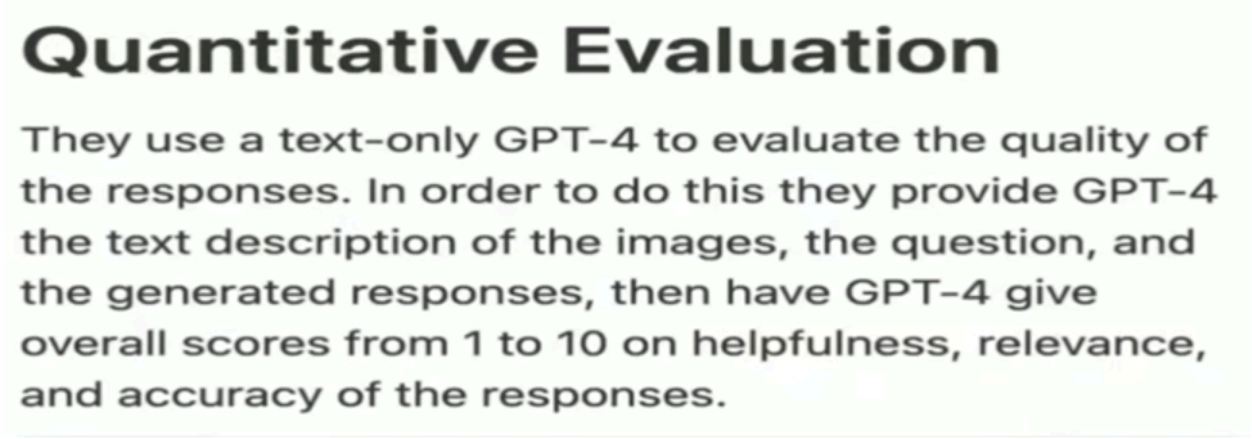
Stage 2: Fine-tuning End-to-End



During the fine tuning they keep the visual encoder weights frozen, and continue to update the projection layer as well as the end to end LLM.

They train the model on 8 x A100s. On this hardware setup they can complete the pre-training on 595k image-text pairs in 4 hours and the fine tuning on 158k conversations in 10 hours.

Stage 3: Quantitative Evaluation



Quantitative Evaluation

They use a text-only GPT-4 to evaluate the quality of the responses. In order to do this they provide GPT-4 the text description of the images, the question, and the generated responses, then have GPT-4 give overall scores from 1 to 10 on helpfulness, relevance, and accuracy of the responses.

LLAVA working:

Model is feeded either a caption or bounding box list of the objects in the images in the context. With these input prompts, we can collect three types of responses -> conversations, detailed descriptions and complex reasoning. The original LLM has been pretrained with all the above pairs

Vision encoder -> Z_v -> projection layer (linear layer) -> H_v -> LLAMA